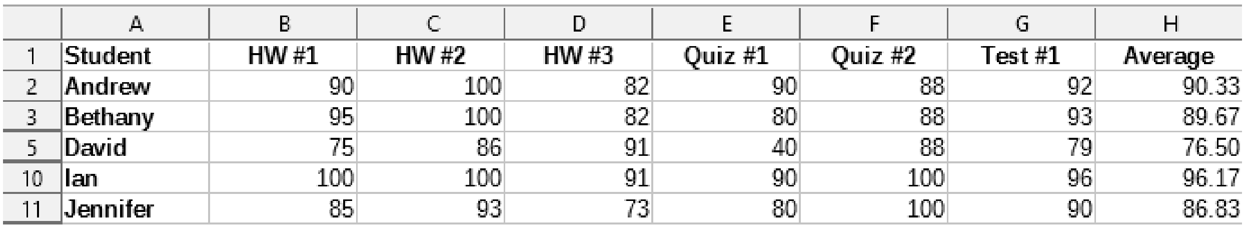
```
  REM CopyOutputData, OutputPosition, and MaxFieldCount.
  oFilterDesc.setFilterFields(oFields())
  oFilterDesc.ContainsHeader = True
  oSheet.filter(oFilterDesc)
End Sub
```



	A	B	C	D	E	F	G	H
1	Student	HW #1	HW #2	HW #3	Quiz #1	Quiz #2	Test #1	Average
2	Andrew	90	100	82	90	88	92	90.33
3	Bethany	95	100	82	80	88	93	89.67
5	David	75	86	91	40	88	79	76.50
10	Ian	100	100	91	90	100	96	96.17
11	Jennifer	85	93	73	80	100	90	86.83

Figure 467: Grading sheet filtered by Quiz #2 values greater than 80

Note

When the *filter* method is called on a sheet, every empty row in the sheet is hidden. When *filter* is called on a range, only empty rows in the range itself are hidden.

Clearing all filters for a sheet

When a filter is applied to a sheet, it replaces any existing filter for that sheet. Therefore, to remove a filter in a sheet, simply create and set an empty filter for that sheet (Listing 25).

Listing 25: RemoveSheetFilter removes the current sheet filter by setting an empty filter

```
Sub RemoveSheetFilter()
  Dim oSheet          ' Sheet to filter.
  Dim oFilterDesc     ' Filter descriptor.

  oSheet = ThisComponent.getSheets().getByIndex(0)
  oFilterDesc = oSheet.createFilterDescriptor(True)
  oSheet.filter(oFilterDesc)
End Sub
```

Filtering multiple columns and filtering with regular expressions

The macro in Listing 26 demonstrates a filter that filters two columns and uses regular expressions. Note that the *filter* method is called on a range rather than its sheet in this example. Figure 468 displays the results of this macro on the grading sheet example in Figure 452.

Listing 26: SimpleRangeFilter uses two columns

```
Sub SimpleRangeFilter()
  Dim oSheet          ' Sheet to filter.
  Dim oRange          ' Range to be filtered.
  Dim oFilterDesc     ' Filter descriptor.
  Dim oFields(1) As New com.sun.star.sheet.TableFilterField
```